

1. Napisati program za uC 8051 koji treba da broji impulse koji se javljaju na ulaznom pinu P3.2 i meri njihovo trajanje u mikrosekundama. Koristiti Timer 0 u modu 1. Definirati maksimalno trajanje impulsa koje se može izmeriti.

a) Koristiti pooling metod - e za spoljni interrupt (bez upotrebe prekidnog potprograma)

b) Koristiti prekidni potprogram za za spoljni interrupt

A)

```
#include <reg52.h>
```

```
/*-----
Timer 0 Overflow Interrupt
-----*/
unsigned int T0_ISR_count = 0;
unsigned int INT0_ISR_count = 0; //max 2^16 -1 puta mozemo da
restartujemo brojac
Unsigned int pom =0;
unsigned long T0_pulswidth_us = 0; // max 2^32 -1 moze biti
trajanje impulsa
```

```
void T0_ISR (void) interrupt 1
{
T0_ISR_count++;
TF0 = 0; // Reset the interrupt request
}
```

```
/*-----
MAIN C function
-----*/
void main (void)
{
```

```
/*-----
Enable interrupts for timer 0.
-----*/
ET0 = 1;
EA = 1;
/*-----
Set Timer0 for 16-bit interval timer
mode.
-----*/
TMOD = (TMOD & 0xF0) | 0x09; // Gate0 = 1, C/T =0; M1 = 0, M1 =1
```

```
while (1)
{
/*-----
Clear the timer overflow counter and
the timer low and high registers. Then,
start the timer.
-----*/
TH0 = 0;
TL0 = 0;
TR0 = 1;
```

```
/*-----
Wait for the pulse to start.
Then, wait for the pulse to end.
-----*/
while (!INT0);
while (INT0);
INT0_ISR_count++;
```

```

/*-----
Compute the width of the pulse -- one
clock cycle is 1us for a standard 8051
and display it. // if another pulse occurs while we're calculating this
then it can disturb our math
-----*/
    pom = TH0 <<8; // učitaj visi bajt
    pom = (pom & 0xFF00) | TL0; // učitaj nizi bajt
    T0_pulswidth_us = 0xFF00 * T0_ISR_count;
    T0_pulswidth_us = T0_pulswidth_us + pom;

}
}

```

B) Potrebno je da se podesi da na pinu INTO imamo visok signal kad se pojavi impuls čije trajanje hocemo da merimo - da bi tad Timer0 počeo da broji a da se externi interapt INTO okida na padajucu ivicu - dakle kad impuls prestaje, kako bismo tad u prekidnom potprogramu mogli da sracunamo vreme trajanja impulsa koji se upravo završio.

```

#include <reg52.h>
unsigned int T0_ISR_count = 0;
unsigned int INTO_ISR_count = 0; //max 2^16 -1 puta mozemo da restartujemo
brojac
Unsigned int pom;
unsigned long T0_pulswidth_us = 0; // max 2^32 -1 moze biti trajanje impulsa

void T0_ISR (void) interrupt 1
{
    T0_ISR_count++;
    TF0 = 0; // Reset the interrupt request
}

void INTO_ISR (void) interrupt 0
{
    INTO_ISR_count++;
    pom = (TH0 <<8) | TL0;
    T0_pulswidth_us = 0xFF00 * T0_ISR_count;
    T0_pulswidth_us = T0_pulswidth_us + pom;
    IE0 = 0; // Reset the interrupt request
    TL0=0x00;
    TH0 = 0x00;
}

void main (void)
{
    EX0 = 1;
    ET0 = 1;
    EA = 1;
    PX0 =1;
    PT0 =1;
    IT1 = 1; //triger INTO na opadajucu ivicu
    TMOD = (TMOD & 0xF0) | 0x09; // Gate0 = 1, C/T =0; M1 = 0, M1 =1
    TH0 = 0x00;
    TL0 = 0x00;
    TR0 = 1; //startujemo timer0 - timer ce krenuti da broji tek kad
signal na INTO ulazu postane logicka 1 bez obzira sto je TR0 = 1

    while (1)
    {

    }
}

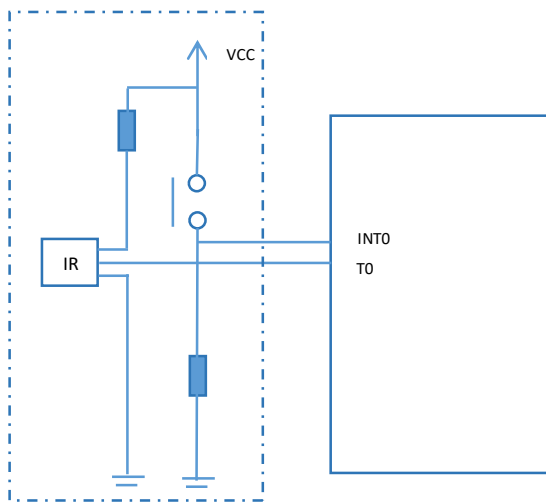
```

}

2. Na raskrsnici sa semaforom je instaliran IC senzor cija je namena detekcija vozila (pri detektovanju objekta na izlazu senzora je stanje logicka 1). Iz semafora je takodje izveden signal koji je aktivan (logicko stanje je logicka 1) za vreme dok traje zeleno svetlo. Definirati vezu sa uC8051 i napisati program za uC 8051 koji treba da registruje broj vozila koja prolaze kroz semafor u toku trajanja zelenog svetla upotrebom Tajmera/brojaca T0.

Objasnjenje:

Signal iz IC senzora treba povezati na pin P3.4 (T0) uC8051 dok je signal koji je aktivan (logicko stanje je logicka 1) za vreme dok traje zeleno svetlo doveden na pin P3.2 (INT0).



A) Bez interupta za INT0

```
#include <reg52.h>
unsigned int vehicle_count = 0;

void main (void)
{
    TMOD = (TMOD & 0xF0) | 0x0E; // Gate0 = 1, C/T =1; M1 = 1, M0 =0
    while (1)
    {
        TLO=0x00;
        TH0=0x00;
        TR0 = 1;
        while (!INT0);
        while (INT0);
        vehicle_count = (vehicle_count & 0x0000) | TLO;
    }
}
```

B) Sa interuppt potprogramom za INT0

```
#include <reg52.h>
unsigned int vehicle_count = 0;

Void extint0_ISR( ) interrupt 0
{
    vehicle_count = (vehicle_count & 0x0000) | TLO;
}
```

```
    TL0=0x00;  
    TH0=0x00;  
    IE0 = 0;  
}
```

```
void main (void)  
{  
    TMOD = (TMOD & 0xF0) | 0x0E; // Gate0 = 1, C/T =1; M1 = 1, M0 =0  
    TL0=0x00;  
    TH0=0x00;  
    TR0=1;  
    while (1)  
    {  
  
    }  
}
```